



## **Rapport de projet de fin de Semestre**

**Thème :**

# **Navigateur Spatio-Temporel**

**Réalisé par :**

- **M.Achraf LANSARI**
- **M.AbdelAli HMESSAR**

## **INTRODUCTION AU PROJET :**

Le projet consiste à développer une interface permettant de trouver des contenus dans la base de patrimoine culturel Europeana suivant des critères géographiques et temporels. Le Navigateur Spatio-temporel est une plateforme basée sur le Web pour l'organisation, la recherche et la visualisation des contenus à partir de données géographiques et temporelles. Permet d'extraire des données des lieux et des dates à partir d'un flux textuel (par exemple flux RSS) et les projeter sur un service de cartographie en ligne (p.ex. Google Maps) et un navigateur temporel (p.ex. MIT Time line ).

# Chapitre

# 1

## I. Navigateur Spatio-Temporel

*Le premier chapitre présente:*

- ✕ *Objectifs du projet*
- ✕ *Entités spatio-temporelles*
- ✕ *Mise au point de la problématique*
- ✕ *A Qui S'adresse L'Interface*
- ✕ *Fonctionnalités de l'application*
- ✕ *Arborescence du projet*
- ✕ *Phases de réalisation du projet*

# **I. Navigateur Spatio-Temporel**

## **I.1 Objectifs du projet**

Développer une application Web pour naviguer dans des contenus du patrimoine mondiale à partir de données géographiques et temporelles, En ajoutant une couche descriptive et sociale qui se décrit en récupérant les tweet correspondant à la recherche ainsi qu'aux données calculées par wolfram, ainsi qu'un affichage des images téléchargés en 3d sous forme d'un musée 3d.

## **I.2 Entités spatio-temporelles**

Une entité spatio-temporelle est une représentation des entités du monde réel composée d'une identité, de propriétés descriptives et de propriétés spatiales. Tandis que l'identité décrit une sémantique fixe de l'entité, les propriétés descriptives et spatiales peuvent varier dans le temps et constituent la partie dynamique de l'entité.

## **I.3 Mise au point de la problématique**

Pour notre projet , la problématique peut être résumée dans les points suivants :

- ✓ Le temps d'obtention des requêtes Ajax
- ✓ Récupérer que les données géolocalisable et qui représente on un laps de temps
- ✓ Surcharge de requêtes, sur la carte Google Maps
- ✓ l'octroi de licences des médias

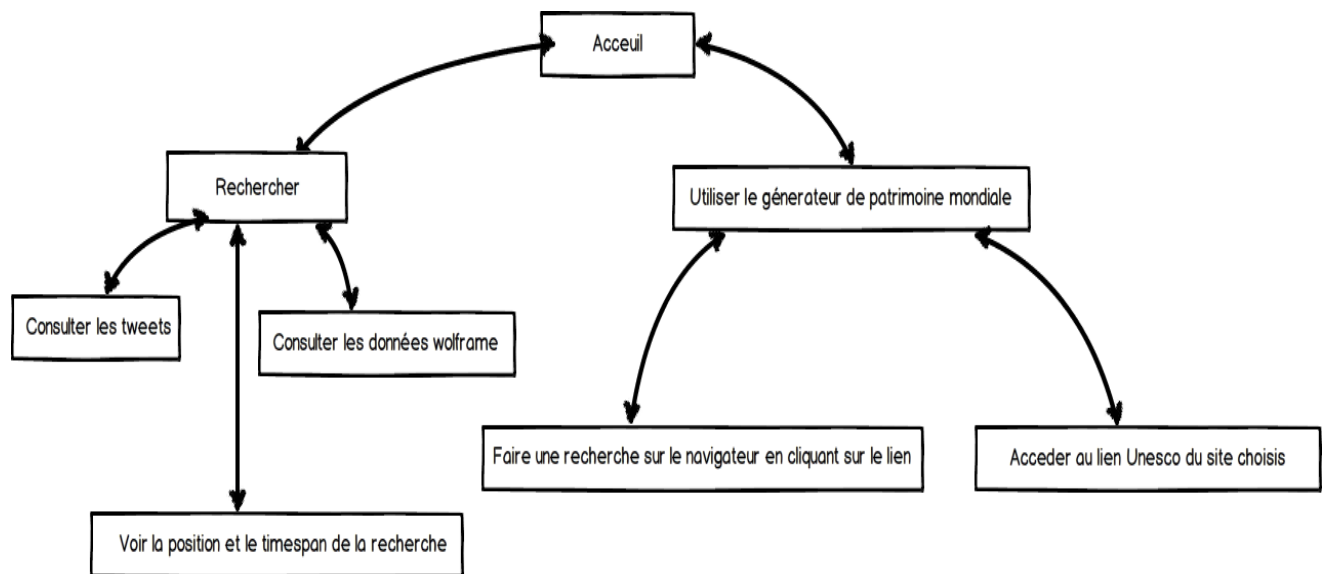
## **I.4 A QUI S'ADRESSE L'INTERFACE**

Les applications spatio-temporelles ont vu le jour pour répondre au besoin de représenter des phénomènes impliquant l'espace et le temps, notre application elle, représente des données du patrimoine mondial, culturel, pour permettre aux utilisateurs de faire des recherches avancée, et de la visualisation.

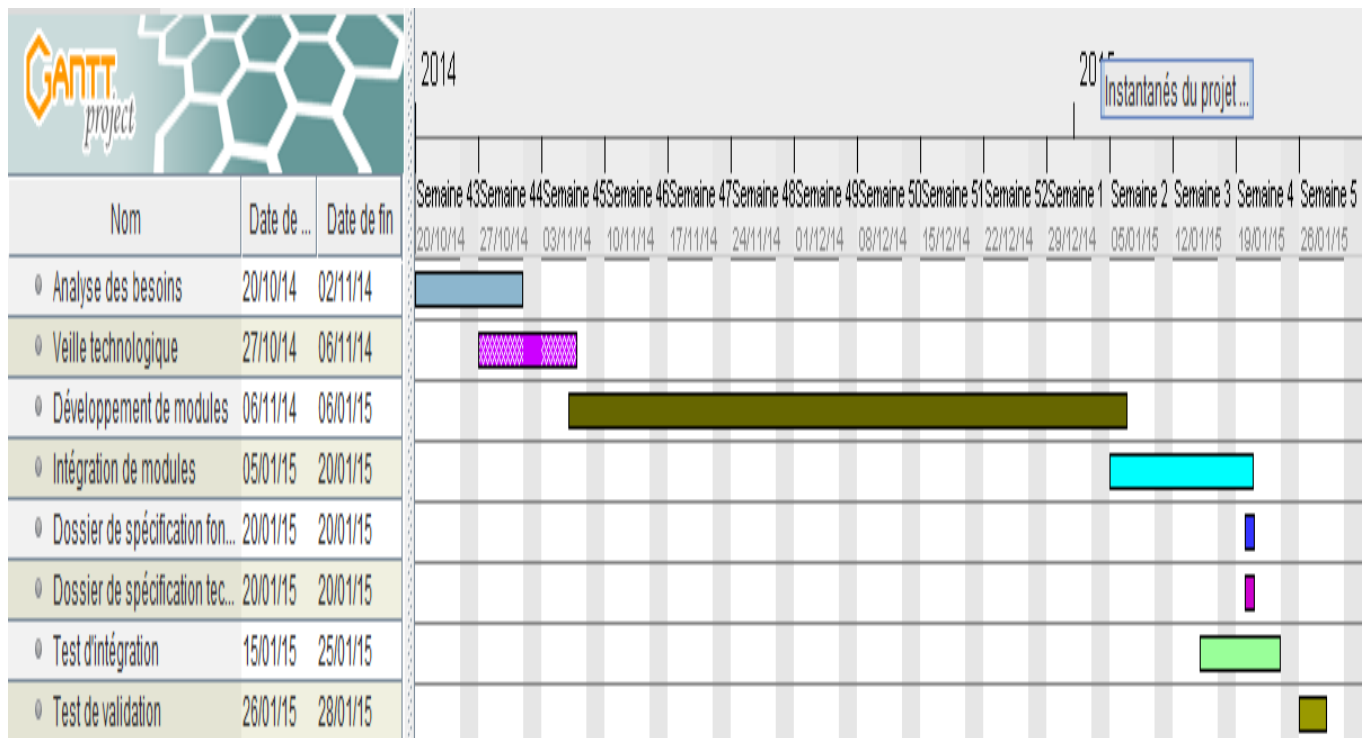
## I.5 Fonctionnalités de l'application

- ✓ Effectuer une recherche selon critères (nombre d'éléments, laps de temps, type, position)
- ✓ Générer la liste de sites du patrimoine mondiale d'un pays en aléatoire
- ✓ Récupération des tweets par rapport à la recherche
- ✓ Récupération des données Wolfram
- ✓ Récupération des données météorologique
- ✓ Affichage des images de patrimoine dans une interface 3d
- ✓ Recherche par laps de temps
- ✓ intervalle modifiable en temps-réel dans « timeline »

## I.6 Arborescence du Projet



## I.7 Phases de réalisation du projet



❖ La phase de conception se réduit à 2 étapes :

- ✓ Analyse de besoin en se basant sur les spécifications fonctionnelles, qui représentent l'ensemble des besoins et les spécifications techniques, qui détaillent comment le produit va être réalisé techniquement.
- ✓ Veille Technologique qui a permet de faire une analyse de l'existant, pour utiliser les technologies, Api nécessaires.

❖ La phase de Développement contient 2 étapes :

- ✓ Développement de tous les modules nécessaires à l'exécution de toutes les fonctionnalités du projet.
- ✓ Intégration des modules développés dans le projet principale en respectant l'ergonomie et accessibilité de l'application.

# Chapitre

# 2

## II. Technologies utilisées

*1-Analyse de l'existant*

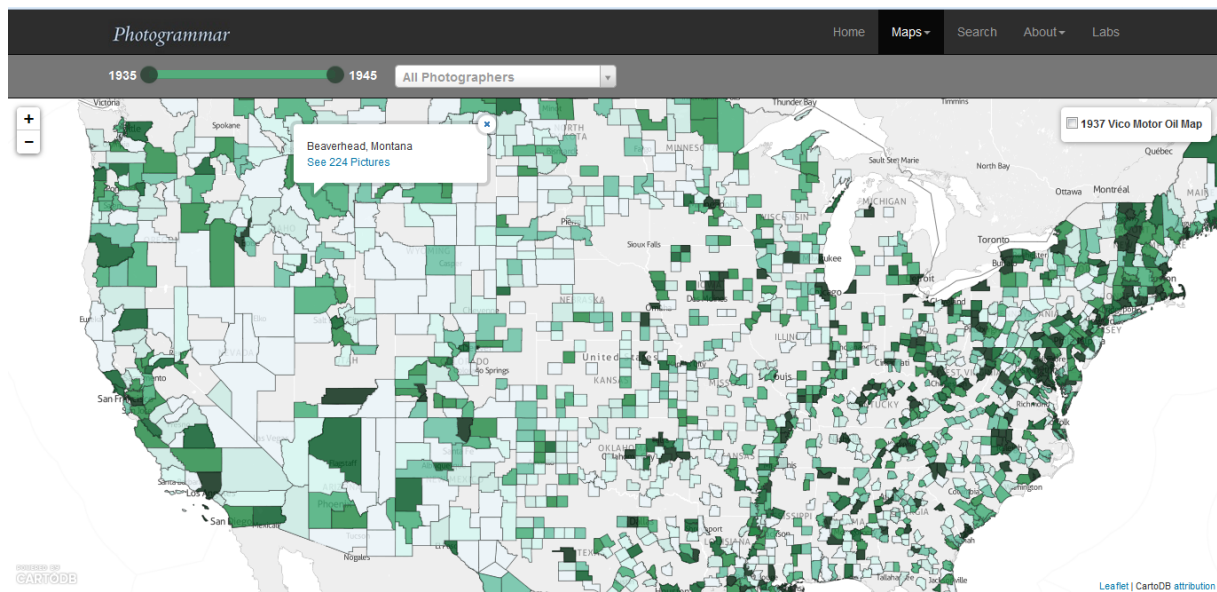
*2-Outils utilisé*

- *API de recherche Twitter*
  - *Comment construire une requête*
  - *Limitation de débit*
  - *Exemples de recherche*
- *L'Europeana API*
- *API Météo*
- *Simile Timeline*

*3-Captures d'Ecran*

## I. Analyse de l'existant :

Les navigateurs spatio-temporel existant se base sur des données statiques (Un Fichier JSON, XML..) Pour gérer l'espace et le temps, ce qui est à notre point de vue une limite d'interactivité comme montre l'exemple ci-dessus, ce que nous proposons est un navigateur spatio-temporel qui utilise des données dynamiques, pour cela on a utilisé les outils suivant :



## II. Outils Utilisé

### II.1 API de recherche Twitter

Le twitter API de recherche fait partie de v1.1 REST API de Twitter. Il permet des requêtes sur les indices de Tweets récents ou populaires et se comporte De manière similaire, mais pas exactement comme la fonction de recherche disponible dans Twitter mobiles ou web clients, tels que la recherche Twitter.com.

Avant de se lancer, il est important de savoir que l'API de recherche est axée sur la pertinence et non l'exhaustivité.

#### II.1.1 Comment construire une requête

La meilleure façon de créer une requête et tester si elle est valide et sera de retour Tweets appariés est d'abord l'essayer au twitter.com/search. Comme vous obtenez un ensemble de résultat satisfaisant, l'URL chargé dans le navigateur contient la syntaxe de requête appropriée qui peut être réutilisé dans le critère d'évaluation de l'API. Voici un exemple:



Nous voulons rechercher pour le compte de tweets. Tout d'abord, nous courons le recherche sur [twitter.com/search](https://twitter.com/search)

Vérifier et copier l'URL chargé. Dans ce cas, nous avons obtenu:  
<https://twitter.com/search?q=%40twitterapi>

.

### **II.1.2 Limitation de débit**

la Twitter API REST 1.1 et est limité taux similaire à d'autres méthodes de v1.1 les utilisateurs représentés par jetons d'accès peuvent faire 180 demandes / requêtes pour 15 minutes.

### **II.1.3 Exemples de recherche**

Lorsque vous suivez un événement qui se passe actuellement, vous seriez intéressé à la recherche de tweets récentes utilisant le hash tag de l'événement, Vous voulez tweets qui contiennent le #superbowl hash tag Votre recherche est URL:

[https://api.twitter.com/1.1/search/tweets.json?q=%23superbowl&result\\_type=recent](https://api.twitter.com/1.1/search/tweets.json?q=%23superbowl&result_type=recent)

## **II.2 L' Europeana API**

REST vous permet de créer des applications qui utilisent la richesse des objets du patrimoine culturel stockés dans le référentiel Europeana. Les utilisations de la technologie API Web standard de REST appelle sur HTTP. Les réponses sont retournées dans le format JSON populaire.

### **II.2.1 Requête**

Chaque appel de l'API est une requête HTTP dans un format spécifié qui est envoyé au service de l'API Europeana.

L'URL API racine est situé au: <http://www.europeana.eu/api/v2>, une authentification est requise. Chaque réponse à un appel d'API contiendra un certain nombre de champs standard qui précèdent les domaines spécifiques de l'appel.

### **II.2.2 Limitation**

On peut effectuer jusqu'à 10 000 appels en 24 heures. Pour plus que cela il faut contacter Europeana pour acheter un abonnement.

## **II.3 API Météo**

API météo (l'interface de programmation d'application) de World Weather ligne permet aux développeurs et programmeurs d'accéder aux données météorologiques actuelles, passées et futures pour une utilisation dans les applications et les sites Web. Notre API météo est facile à utiliser et offre de pointe, des informations détaillées météo. Ces données sont livrées à l'aide de requêtes HTTP standard, avec notre API météo retourné dans des formats standards de l'industrie tels que XML, JSON et JSON-P.

## **II.4 Simile Timeline :**

Timeline permet aux créateurs de sites Web à incorporer un système de chronologie interactive dans leurs sites. Elle exige seulement du Javascript. Il est souvent désigné comme "Google Maps" pour le temps. On peut rajouter des bandes de chronologies selon les heures, jours, mois ou années, spécifier des intervalles ou ajouter des événements.

## **3.2. Choix technologiques**

### **3.2.3.JQuery :**

Nous avons choisi d'utiliser la bibliothèque JavaScript jQuery, elle présente plusieurs avantages:

Elle permet de programmer beaucoup plus rapidement une fonctionnalité que de l'écrire à partir de rien. Elle permet donc de rehausser la valeur des projets de conception Web en diminuant le temps consacré ou bien en permettant d'implémenter des fonctionnalités plus poussées,

Ensuite, elle est conçue pour fonctionner dans tous les navigateurs récents (ainsi qu'Internet Explorer 6 à 8), le fait d'utiliser une librairie aussi poussée permet d'éviter d'avoir recours à Flash.

### **3.2.4. Ajax:**

En utilisant Ajax, le dialogue entre le navigateur et le serveur se déroule la plupart du temps de la manière suivante : un programme écrit en langage de programmation JavaScript, incorporé dans une page web, est exécuté par le navigateur. Celui-ci envoie en arrière-plan des demandes au serveur Web, puis modifie le contenu de la page actuellement affichée par le navigateur Web en fonction du résultat reçu du serveur, évitant ainsi la transmission et l'affichage d'une nouvelle page complète. Ce qui nous permet de ne pas réactualiser la page, ainsi que d'exécuter plusieurs tâches en simultanée et en arrière-plan, sans que l'utilisateur ne se rende compte pour augmenter la réactivité de l'application Web.

# Chapitre

# 3

## III. Conclusion

*1-Limites du projet*

*2-Evolutions du projet*

*1-Conclusion*

*2-Bibliographie*

## **I. Limites du projet**

### **I.1 Limitations Api**

Comme décrits dans la partie Api utilisés chaque Api a sa propre limitation par rapport au nombre de requêtes passé par heures, jours ou mois.

### **I.2 Limitations d'entités spatio-temporel**

Tous les objets ne sont pas représentant d'une entité spatio-temporel, l'un ou les deux axes sont manquant qui veut dire qu'on exécute une recherche avancée pour trouver les données avec des méta data correct ,

### **I.3 Limitations de la géo visualisation HTML5**

Bien que le cadre proposé peut efficacement visualiser les données spatio-temporelle avec les navigateurs courants, il y a encore quelques problèmes avec approche basée sur HTML5. Premier problème est en charge du navigateur. Seuls les navigateurs modernes prennent en charge HTML5 qui est un problème pour les gens utilisent encore ancienne version de IE.

### **I.4 Limitations du serveur**

Il faut penser à optimiser toutes les requêtes et vider la mémoire, à chaque nouvelle requête Pour ne pas surcharger le serveur.

## **II. Evolutions du projet :**

- Recherche Avancée en utilisant des types comme : recherche de personnes, recherche par place « where= » », recherche par type par exemple (TYPE:IMAGE), recherche par timestamp en incluant les mois, jours et heures.
- Effectuer un cadrage sur la carte Google maps pour récupérer les positions dans un carré comme exemple et exécuter la requête correspondante
- Gérer une session d'authentification pour garder l'historique des requêtes précédentes ainsi que les images sélectionné pour le musée
- Réaliser un graphe de statistique pour représenter le changement par rapport a une recherche

### **III. Conclusion :**

Notre Navigateur Spatio-Temporel est un des seuls à être dynamique, permettant de faire n'importe quelle recherche on veut, La couche sociale est descriptive permet d'avoir une vue différentes sur les données voulus, la problématique de gestion des licences de médias est respectés par Europeana, tous les médias disposent de leur licences.

Avec les évolutions cités et l'optimisation des requêtes, Le projet sera puissant et utile

Dans un but de recherche spatio -temporel et de visualisation.

## **Bibliographie :**

[fr.wikipedia.org/wiki/API](http://fr.wikipedia.org/wiki/API)

<https://dev.twitter.com/>

[products.wolframalpha.com/api/](http://products.wolframalpha.com/api/)

<http://www.europeana.eu/portal/api-working-with-api.html>

<http://www.simile-widgets.org/timeline/>

<http://www.worldweatheronline.com/>

<http://www.dhteumeuleu.com>

[tympanus.net/codrops](http://tympanus.net/codrops)